C. AMENDMENTS TO THE SPECIFICATION

Please replace the paragraph beginning on page 1 at line 7 with the following paragraph:

In order to process large numbers of secure electronic transactions, organizations deploy systems consisting of multiple identical cryptographic devices. As used herein, the term cryptographic device means a–an electrical or electronic contrivance with the purpose of performing one or more cryptographic operations. A cryptographic device may be hardware such as programmable card, or it may be a computer with software instructions for executing the cryptographic operations. A card is an electronic circuit board that is plugged into a slot in a system unit. A cryptographic operation is an action that changes data in order to set up encryption, to perform encryption, to perform decryption, and to verify the accuracy of data. As used herein, cryptographic operations include but are not limited to key generation, encryption, decryption, hash operations and digital signature generation and verification. In order to increase capacity for processing large numbers of cryptographic operations, additional identical cryptographic devices may be added to the system. Cryptographic operations vary significantly in the amount of time required to complete a particular type of operation. A need exists for a way to distribute incoming requests for cryptographic operations among multiple cryptographic devices so that maximum utilization of the devices is achieved.

Please replace the paragraph beginning on page 5 at line 10 with the following paragraph:

An operating system runs on processor **202** and is used to coordinate and provide control of various components within data processing system **200** in Figure 2. The operating system may be a commercially available operating system such as OS/2, which is available from International Business Machines Corporation. "OS/2" is a trademark of International Business Machines Corporation. An object oriented programming system, such as Java, may run in conjunction with the operating system and provides calls to the operating system from Java programs or applications executing on data processing system **200**. "Java" is a trademark of Sun Microsystems, Inc. Instructions for the operating system, the object-oriented operating system, and applications or programs may be located on storage devices, such as hard disk drive **226**, and they may be loaded into main memory **204** for execution by processor **202**. Those of ordinary skill in the art will appreciate that the hardware in Figure 2 may vary depending on the implementation. Other internal hardware or peripheral devices, such as flash ROM (or equivalent nonvolatile memory) or optical disk drives and the like, may be used in addition to or in place of the hardware depicted in Figure ~~3~~2. Also, the processes of the present invention may be applied to a multiprocessor data processing system. For example, data processing system **200**, if optionally configured as a network computer, may not include SCSI host bus adapter **212**, hard disk drive **226**, tape drive **228**, and CD-ROM **230**, as noted by the box with the dotted line in Figure 3 denoting optional inclusion. In that case, the computer, to be properly called a client computer, must include some type of network communication interface, such as LAN adapter **210**, modem **222**, or the like. As another example, data processing system **200** may be a stand-alone system configured to be bootable without relying on some type of network communication interface, whether or not data processing system **200** comprises some type of network communication interface. As a further example, data processing system **200** may be a Personal Digital Assistant (PDA) device which is configured with ROM and/or flash ROM in order to

provide non-volatile memory for storing operating system files and/or user-generated data. The depicted example in Figure 2 and above-described examples are not meant to imply architectural limitations with respect to the present invention. It is important to note that while the present invention has been described in the context of a fully functioning data processing system, those of ordinary skill in the art will appreciate that the processes of the present invention are capable of being distributed in a form of a computer readable medium of instructions and a variety of forms and that the present invention applies equally regardless of the particular type of signal bearing media actually used to carry out the distribution. Examples of computer readable media include recordable-type media, such a floppy disc, a hard disk drive, a RAM, and CD-ROMs, and transmission-type media, such as digital and analog communications links.

Please replace the paragraph beginning on page 8 at line 2 with the following paragraph:

Referring to Figure 4, a block diagram depicts data processing system **400**, which may be implemented as a server, such as server **304** in Figure 3, in accordance with the present invention. Data processing system **400** may be a symmetric multiprocessor (SMP) system including a plurality of processors **402** and **404** connected to system bus **406**. Alternatively, a single processor system may be employed. Also connected to system bus **406** is memory controller/cache **408**, which provides an interface to local memory **409**. I/O bus bridge **410** is connected to system bus **406** and provides an interface to I/0 bus **412**. Memory controller/cache **408** and I/0 bus bridge **410** may be integrated as depicted. Peripheral component interconnect (PCI) bus bridge **414** connected to I/O bus **412** provides an interface to PCI local bus **416**. Modem **418** may be connected to PCI bus **416**. Typical PCI bus implementations will support four PCI expansion slots or add-in connectors. Communications links to a network may be provided through modem **418** and network adapter **420** connected to PCI local bus **416** through add-in boards. Additional PCI bus bridges **422** and **424** provide interfaces for additional PCI buses **426** and **428**. Cryptographic devices (Crypto) 1 through 8 are attached. Crypto 1 **425**, Crypto 2 **428**, Crypto 3 **430** and Crypto 4 **432** are attached to PCI bus **4232**. Crypto 5 **434**, Crypto 6 **436**, Crypto 7 **438** and Crypto 8 **440** are attached to PCI bus **425**. A hard disk **432** may also be connected to I/O bus **412** as depicted, either directly or indirectly. Those of ordinary skill in the art will appreciate that the hardware depicted in Figure 4 may vary. For example, other peripheral devices, such as optical disk drive and the like, also may be used in addition or in place of the hardware depicted. The depicted example is not meant to imply architectural limitations with respect to the present invention. The data processing system depicted in Figure 4 may be, for example, an IBM RISC/System 6000 system, a product of International Business Machines Corporation in Armonk, New York, running the Advanced interactive Executive (AIX) operating system.

Please replace the paragraph beginning on page 18 at line 18 with the following paragraph:

Fig. 10 depicts a flow chart for the process of computing T(N). The program begins (**1002**) and N is set to equal 1 (**1004**). Next, the program queries device N (**1006**). In other words, when N is equal to 1, device 1 will be queried. T(N) is then set to equal 0 (**1008**). X is set equal to 1 (**1010**). Queue Item X is queried (**1012**). In other words, when X is set equal to 1, queue item 1 will be queried. Next, ET is set equal to the estimated time value for queue item X (**1014**). T(N) is then set equal to T(N) plus ET (**1016**). Next, a determination is made as to whether or not there is another task in queue (**1018**). If there is another task in queue, X is set equal to X plus 1, and the program goes to step **1012**. If there is not another task in queue, the program proceeds to the next step which is to save T(N) (**1022**). A determination is made as to whether or not there is another device to query (**1024**). If there is another device to query, N is set equal to N plus 1 (**1026**). If there is not another device to query, the program will stop (**1028**).